Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas

Prof. Edécio Fernando Iepsen

Desenvolvimento de Serviços e APIs

Trabalho #3: Adicionar Recursos de Segurança em APIs

Acrescentar recursos de segurança na API desenvolvida no Trabalho #2. Inicialmente adicione os recursos trabalhados em aula, ou seja:

- Criar a model Usuario (contendo: id, nome, email, senha, …)

- Adicionar algum relacionamento desta Model Usuario com outra Model do seu sistema.

- Criar as rotas e as rotinas para realizar a inclusão e listagem dos dados dos usuários.

- Criptografar a senha do usuário.

- Validar a senha, a partir de regras de composição dos caracteres da senha (como, por exemplo, que a senha tenha, no mínimo 8 caracteres, tenha letras minúsculas, maiúsculas, números e símbolos). Impedir a inclusão de um usuário, com uma senha que não contemple essas regras.

- Criar rota de Login com a geração de token. Definir middleware de verificação do token e adicioná-lo em 2 ou 3 rotas do sistema.

- Criar a Model / tabela de Logs (relacionada com a tabela de usuários). Registrar 2 ou 3 ações (ou tentativas de ações) do sistema nos logs.

- Acrescentar as rotas para realização de backup e restore das tabelas do sistema.

Escolher e implementar 2 novos recursos relacionados aos controles de segurança – como, por exemplo:

1. Implementar rotina para recuperação de senha para usuários que esqueceram a senha. O recurso deve ser implementado a partir de 2 rotas/rotinas. A primeira, é para o usuário solicitar a recuperação de senha. Neste processo validar o e-mail e gerar um código com 4 caracteres (por exemplo) e enviar para o e-mail do usuário. A segunda, deve receber e-mail, o código (anteriormente enviado) e a nova senha. Validar e realizar a alteração da senha.
2. Acrescentar na model de usuário os atributos status (que deve iniciar INATIVO, quando ele se cadastra) e um número randômico. No cadastro, enviar para o e-mail do usuário um link (com o número randômico no path) que quando acionado deve alterar o status do usuário para ATIVO.
3. Definir níveis de acesso no cadastro do usuário, onde o usuário – a partir do seu nível, tenha privilégios diferentes no acesso aos recursos do sistema. Testar nas rotas estes níveis (para realizar a exclusão de dados da tabela principal, o usuário tem que ser nível 3, por exemplo).
4. Implementar um controle de limite de tentativas de acesso inválidas para o usuário. Desta forma, ao atingir, por exemplo, 3 tentativas inválidas bloqueia o usuário (não permite novos acessos até ser retirado o bloqueio).
5. Registrar data/hora do último login do usuário. Exibir essa data/hora no login (“Bem-vindo ... Seu último acesso ao sistema foi ...” ou “Bem-vindo. Este é o seu primeiro acesso ao sistema”)
6. Permitir a troca da senha (no caso do esquecimento) a partir de uma outra forma – como, por exemplo, o cadastro de uma pergunta e resposta do usuário no momento do seu cadastro. Na rota/rotina de solicitação de troca, verificar se a resposta está correta e realizar a alteração.
7. Implementar rotina de alteração simples de senha do usuário, validando a senha atual e criptografando a nova senha.

8. Implementar rotina de *Soft Delete* na tabela principal do seu sistema – ou seja, não excluir fisicamente o registro, mas, sim, alterar o atributo deleted deste registro para true (acrescentar também em deletedAt com a data da exclusão) . Não retornar os registros deleted nas listagens.

- Data da Entrega/Apresentação: **04/07/2025**

Conceitos:

- Rotas e funções dos cadastros em funcionamento e os 2 recursos adicionais de segurança implementados corretamente: A

- 1 dos recursos extras ausentes: B

- Sem os recursos extras: C